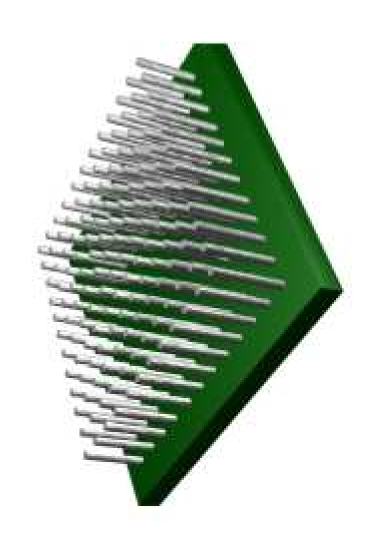
MANUAL DE INSTALACION



Antes de Comenzar

Introducción

SimuProc es una aplicación desarrollada en C++ y ha sido compilada para ser un archivo ejecutable *standalone*, lo que quiere decir que no requiere de programas o librerías externas para su funcionamiento.

Requisitos

- SimuProc correrá en un computador que corra el sistema operativo Microsoft Windows en cualquiera de las siguientes versiones: 95b/98/ME/2000/2003/XP/Vista
- Mínimo 2 MB de espacio libre en Disco
- Se recomienda 64MB de RAM o mas.

Licencia

SimuProc es Freeware, esto quiere decir que es un software que se puede usar, copiar y distribuir libremente pero que no incluye archivos fuentes. El software freeware no se debe pagar para adquirirlo o utilizarlo.

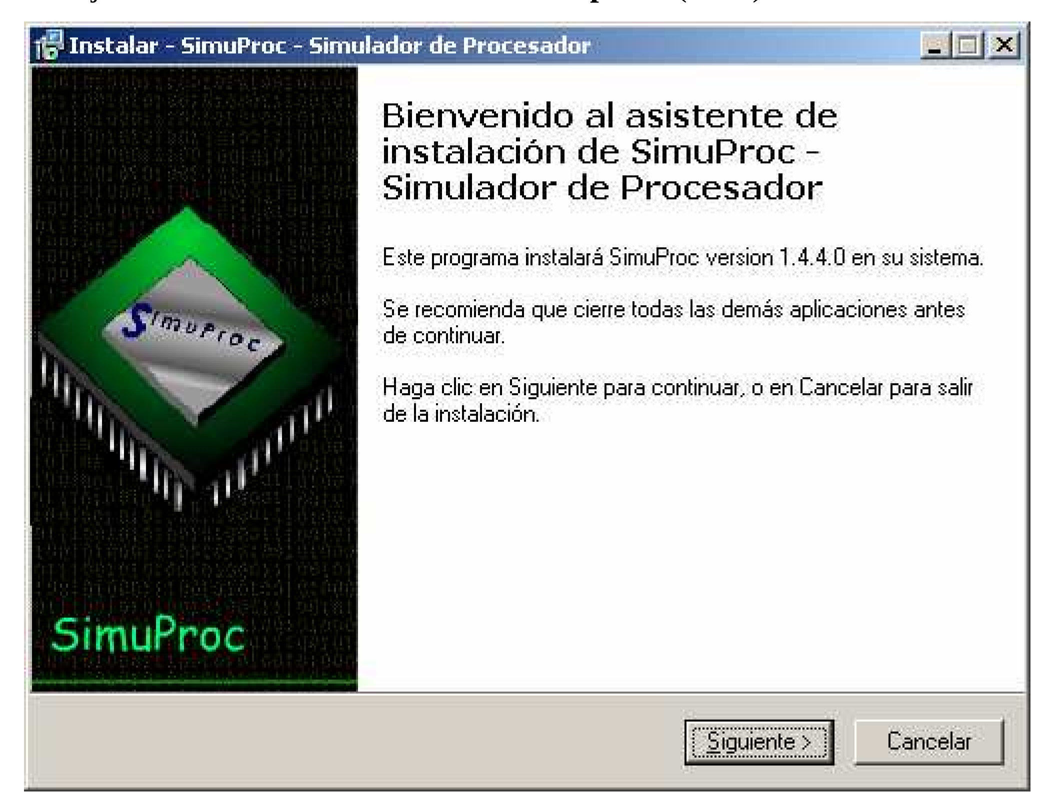
Conseguir

Se puede obtener la ultima versión de SimuProc desde su sitio Web en http://simuproc.tk/ y en caso de no estar arriba, debido a la buena acogida en Internet, solo basta con ingresar su nombre en un motor de búsqueda para encontrar algún sitio alterno de donde descargarlo.

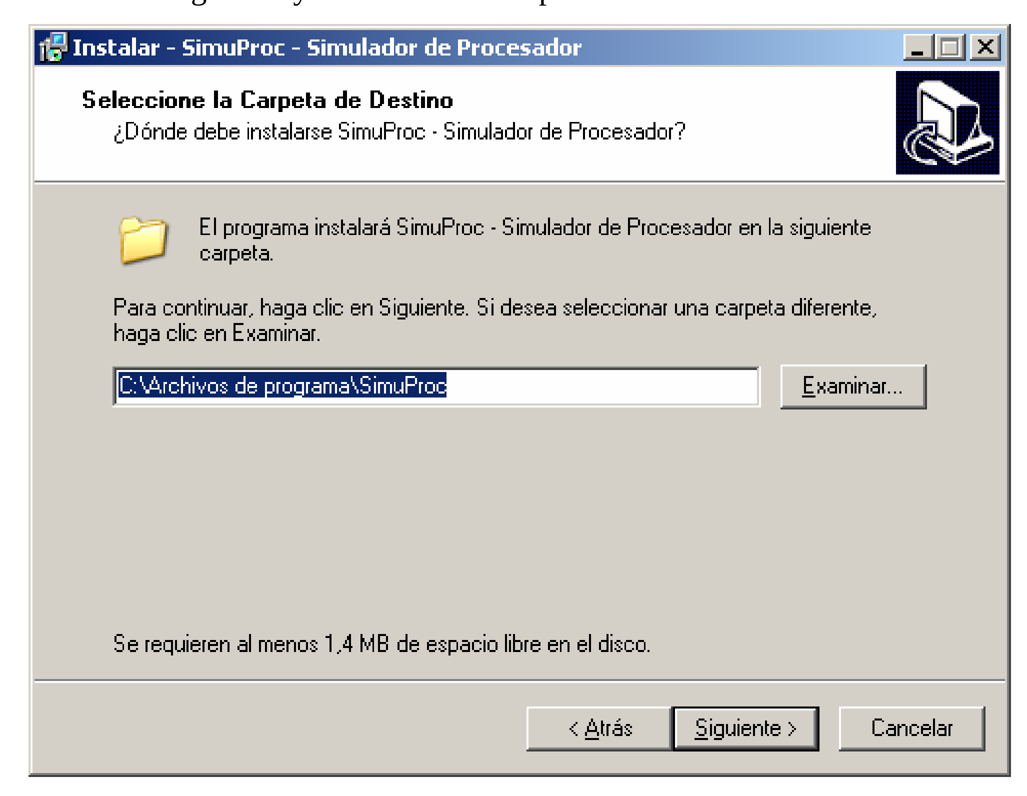
Instalación

SimuProc cuenta con un asistente de instalación el cual hace que todo sea muy sencillo y la aplicación se instala en pocos pasos:

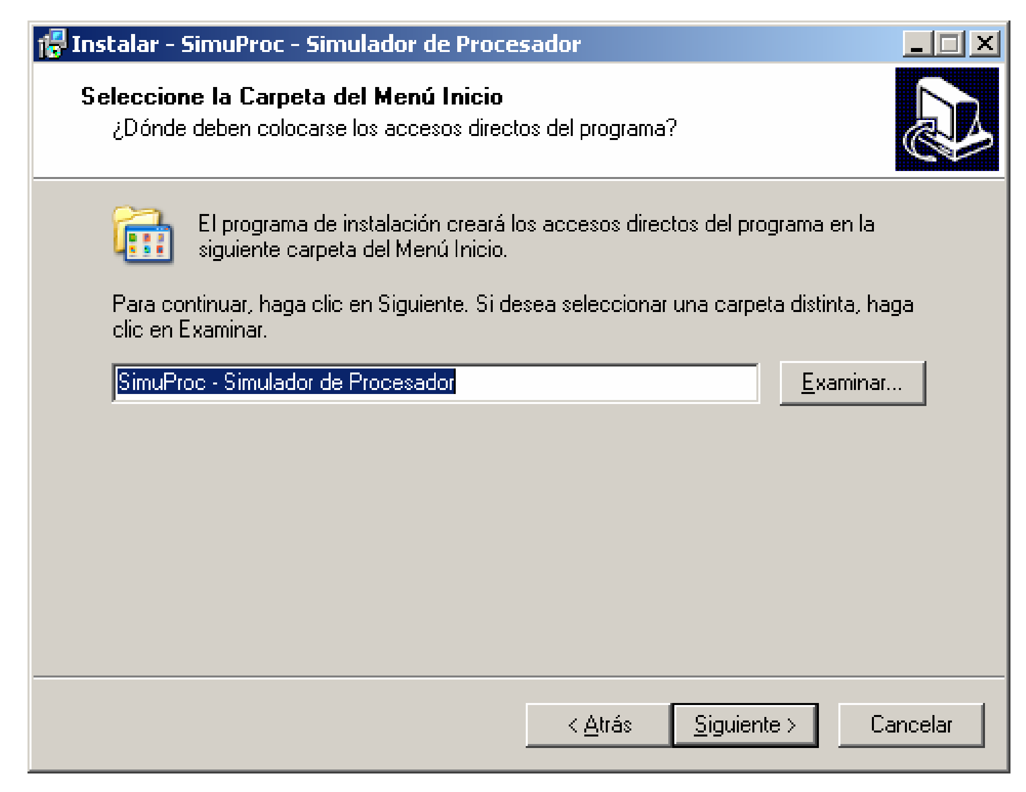
1. Ejecutar el instalador **SimuProc14-Setup.exe** (1MB)



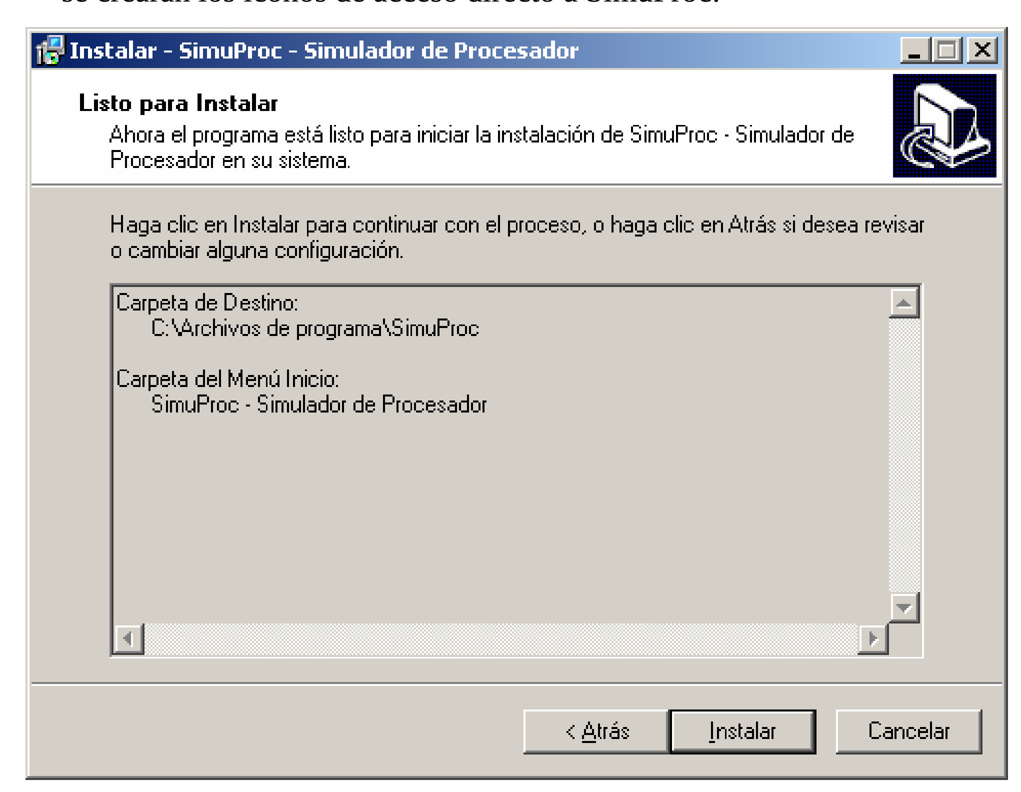
2. Clic en Siguiente y seleccionar la carpeta donde se instalará SimuProc:



3. Clic en Siguiente y escribir el nombre de la carpeta del menú de inicio donde se crearán los iconos de acceso directo a SimuProc:



4. Clic en Siguiente y escribir el nombre de la carpeta del menú de inicio donde se crearán los iconos de acceso directo a SimuProc:



5. Si ya se seleccionó todo como lo querido, ya se está listo para empezar la instalación automática, clic en Instalar y luego de unos segundos ya habrá terminado la instalación:



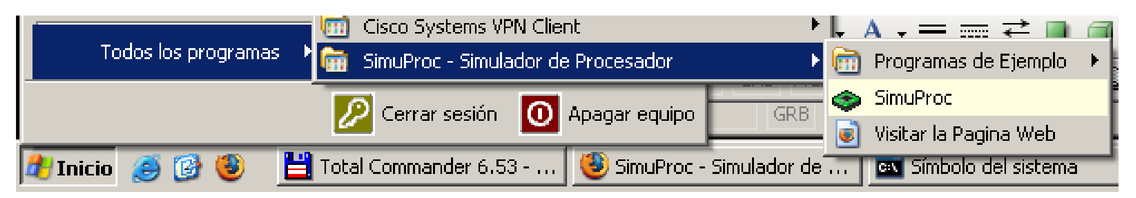
MANUAL DE USUARIO

Introducción

Este manual pretende contener la información necesaria para el usuario de SimuProc. Se explicará cómo usar la interfaz, qué hace cada botón y qué necesitan para funcionar, cómo acceder a los diferentes tipos de posibilidades que ofrece el simulador, etc.

Ejecución

Para ejecutar SimuProc se entra al menú de inicio, buscar la aplicación en las carpetas de menú de inicio y seleccionar SimuProc.

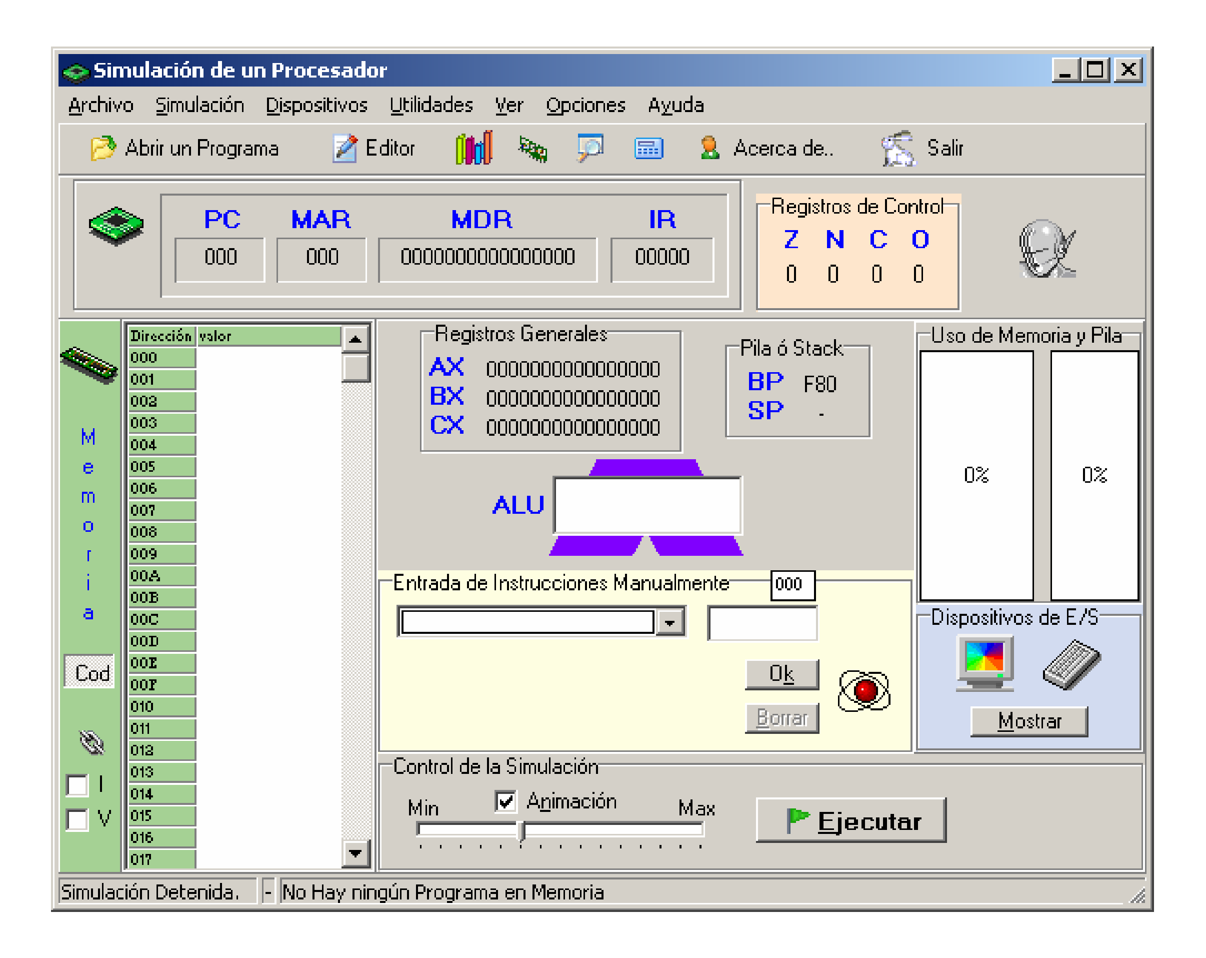


Lo cual ejecutará y cargará la aplicación en memoria, en esta pantalla de inicio aparece el proceso de carga:



Ventana Principal

La interfaz de SimuProc es muy facil de usar, en ella tenemos los diferentes elementos que componen el procesador simulado y con muchos de ellos podremos interactuar directamente.



La barra de herramientas



Esta barra ofrece un acceso mas rápido a las diferentes opciones y ventanas que también se encuentran en el menú.

- Abrir un programa: Aparece un cuadro de dialogo donde se puede cargar un programa que haya sido salvado previamente en formato SMP el cual es un formato propietario de simuproc compatible con las versiones anteriores.
- Editor: Abre la ventana de edición de programas la cual más adelante se describe su funcionamiento.
- Estadísticas de la Simulación: Muestra la ventana de estadisticas.
- Modificar Memoria: abre una ventana que permite modificar una posición de memoria la cual permite especificar la dirección que se desea modificar.

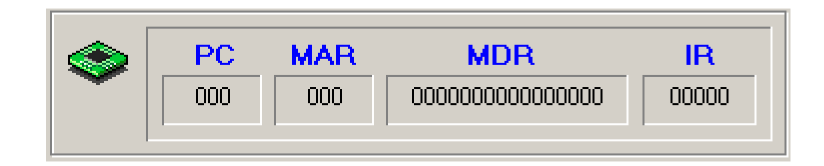


- Vigilante de Memoria: Abre el dialogo de vigilante de memoria.
- Conversión de Bases: Abre el dialogo de conversión de bases.
- Acerca De: Muestra información de la aplicación tal como versión, autor del programa, etc.



- Salir: Sale del simulador. Si se tiene en memoria un programa que no ha sido salvado se le solicitará si desea guardar los cambios.

Registros apuntadores



PC ó IP: Program Counter o Instruction Pointer, Contiene la dirección de memoria de la próxima instrucción a ejecutar y es incrementado en cada nueva instrucción.

MAR: Memory Address Register. (Registro de Dirección de Memoria) es el registro en el que se almacena la dirección de memoria a la que se quiere acceder.

MDR: Memory Data Register o Memory Buffer Register, es un registro intermedio en el que se almacenan los datos que se escriben o leen de memoria. En el caso de una lectura, se pone en el MAR la dirección y se activa la señal de leer, obteniendo en el MDR el dato buscado. En el caso de una escritura, se pone en el MAR la dirección y en el MDR el dato a escribir en memoria, después de activa la señal de escribir, de esta forma almacenamos en memoria el dato.

IR: Instruction Register, en este registro se introduce la instrucción a ejecutar, después de haberla leído de memoria accediendo a ella mediante la dirección señalada en el PC; El contenido de este registro se puede dividir en código de operación (el código que señala la operación que se realizará) y operándos. Puede haber 2 operándos o sólo uno. Acá es donde se decodifica e interpreta la instrucción así: se descompone la instrucción leída de forma que se pueda saber cual es la operación que se desea realizar y cuales son los operándos, en su caso, o el desplazamiento en caso de que se trate de una instrucción de bifurcación.

Registros de control

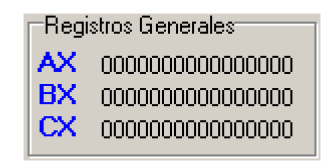


Estos registros se usan para poder controlar el comportamiento de un programa los cuales se activan después de cada operación, según sea el resultado de la instrucción ejecutada.

Zero flag: se vuelve 1 si el resultado de la ultima operación = 0
Negative ó Sign flag: Se vuelve 1 si el resultado de la ultima operación es igual a un numero negativo.
Carry flag: se activa cuando la operación realizada ha producido un acarreo.
Overflow flag: se activa cuando la operación produjo desbordamiento (overflow) , es decir, el resultado ocupaba más de los 16 bits que caben en un registro.

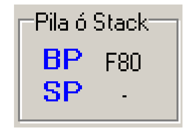
Durante la ejecución de un programa podemos ejecutar hacer doble clic en cualquiera de los registros para invertir su estado y así alterar la ejecución del programa a gusto del programador, ya sea para hacer pruebas o diversión.

Registros generales



Son registros de propósito general los cuales el programador puede usar a su gusto dependiendo de sus necesidades. Inicialmente estos registros se hicieron con un fin, como Acumulador, Base y Contador respectivamente.

Registros de pila

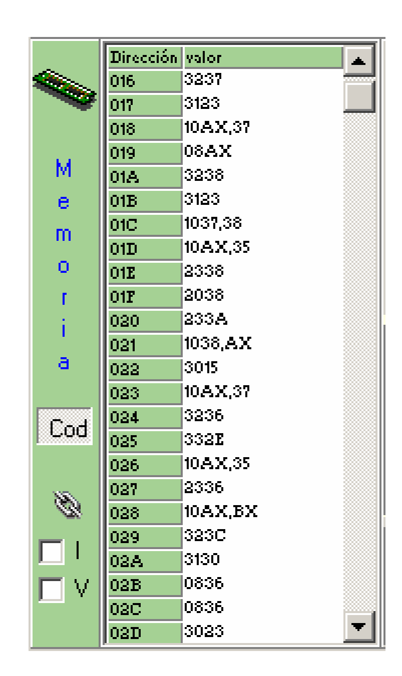


BP: Base Pointer, Puntero de base de la pila. El valor de por defecto es F80 , Este puede cambiarse desde un programa, asignándole otra dirección de memoria con la instrucción MOV. Supóngase que se quiere reservar mas espacio para la pila haciendo que esta comience desde la posición CF1, entonces se copia esta dirección en cualquier posición de memoria; por ejemplo, la dirección 3B entonces se usa la instrucción MOV BP,3B y así BP es igual a CF1. Mientras se ejecuta el programa se puede visualizar en una barra de porcentaje el uso de la pila.

SP: Stack Pointer, Puntero de la pila, indica en que próxima dirección de la pila esta disponible, es decir, apunta a la cima de la pila. Este valor se cambia automáticamente cuando se usan las instrucciones PUSH ó POP.

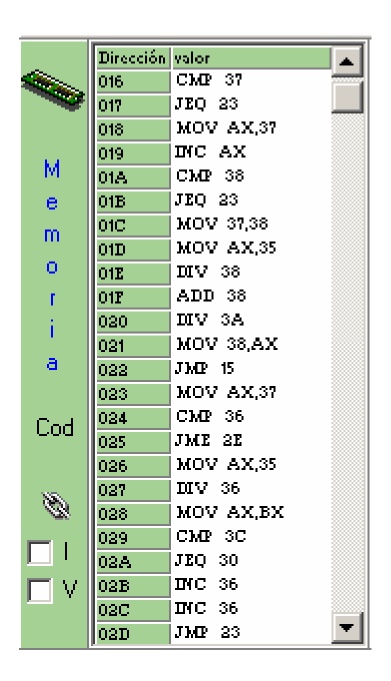
La memoria

En la memoria se puede ver todo el contenido de esta en cada una de sus posiciones. Este contenido puede ser visualizado de dos formas las cuales se pueden intercambiar presionando el botón Cod que hace que nos muestre las instrucciones por código de operación:



Dirección	valor
016	3237
017	3123
018	10AX,37
019	08AX
01A	3238
01B	3123
01C	1037,38
01D	10AX,35
01E	2338
01F	2038
020	233A
021	1038,AX
022	3015
023	10AX,37
024	3236
025	332E
026	10AX,35
027	2336
028	10AX,BX
029	323C
02A	3130
02B	0836
02C	0836
02D	3023

ó de forma nemotécnica que es como los humanos las entendemos:

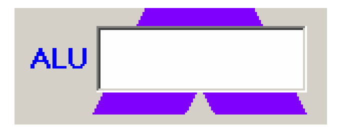


Dirección	valor
016	CMP 37
017	JEQ 23
018	MOV AX,37
019	INC AX
01A	CMP 38
01B	JEQ 23
01C	MOV 37,38
01D	MOV AX,35
01E	DIV 38
01F	ADD 38
020	DIV 3A
021	MOV 38,AX
022	JMP 15
023	MOV AX,37
024	CMP 36
025	JME 2E
026	MOV AX,35
027	DIV 36
028	MOV AX,BX
029	CMP 3C
02A	JEQ 30
02B	INC 36
02C	INC 36
02D	JMP 23

Adicionalmente se dispone de dos cuadros de selección los cuales sirven para que durante una simulación se siga el acceso a memoria en todo momento,

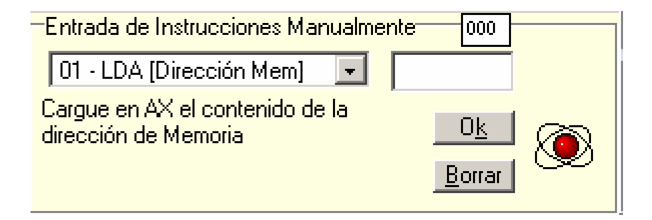
independientemente de las instrucciones con el cuadro I, y de las Variables con el cuadro V, además de poderlo hacer para ambas.

ALU



Aquí se observan los valores con los cuales se esta realizando alguna operación aritmética.

Entrada de instrucciones manualmente



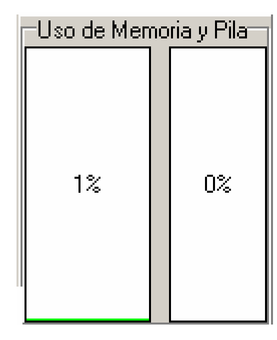
Esta funcionalidad se usaba principalmente en las primeras versiones de SimuProc que no contaba con el editor interno que tiene a partir de la versión 1.3, y servia para ingresar los programas a la memoria, ahora ya no es necesario usarlo pero se deja a discreción del programador.

Dispositivos de E/S



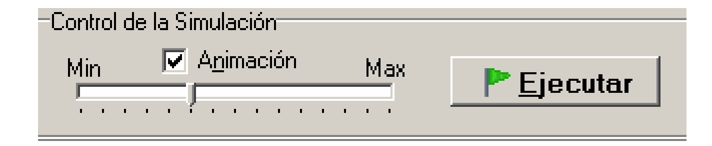
Presionando el botón Mostrar, se obtiene una ventana que será la comunicación de entrada y salida con el usuario cuando se están ejecutando los programas.

Uso de memoria y pila



En estas dos graficas muestran el porcentaje de uso de memoria y pila de acuerdo a su capacidad reservada la cual sirve para detectar un problema de llenado de memoria.

Control de la simulación

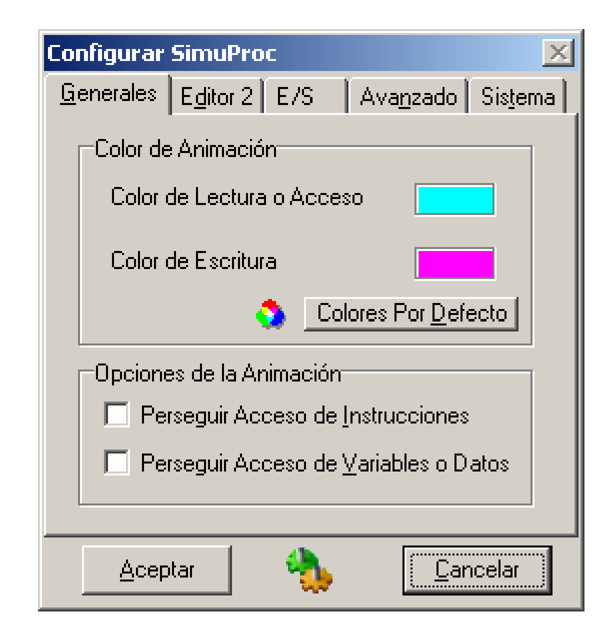


De este control se inicia/pausa la simulación en cualquier momento presionando el botón ejecutar, adicionalmente la velocidad de la simulación y si se anima ó no.

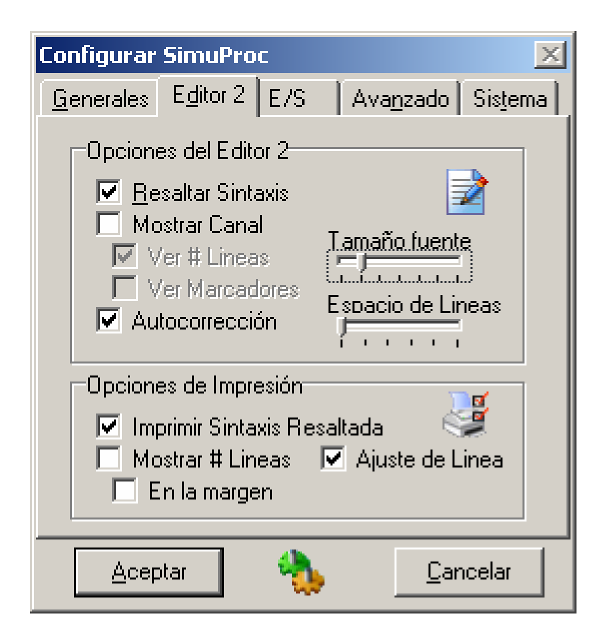
Configurar SimuProc

SimuProc en cierto aspecto permite cambiar algunas de sus configuraciones por defecto a nuestro gusto, como por ejemplo el color de la animación de lectura y escritura.

Las dos opciones presentadas previamente en la memoria, también son accesibles desde aquí.

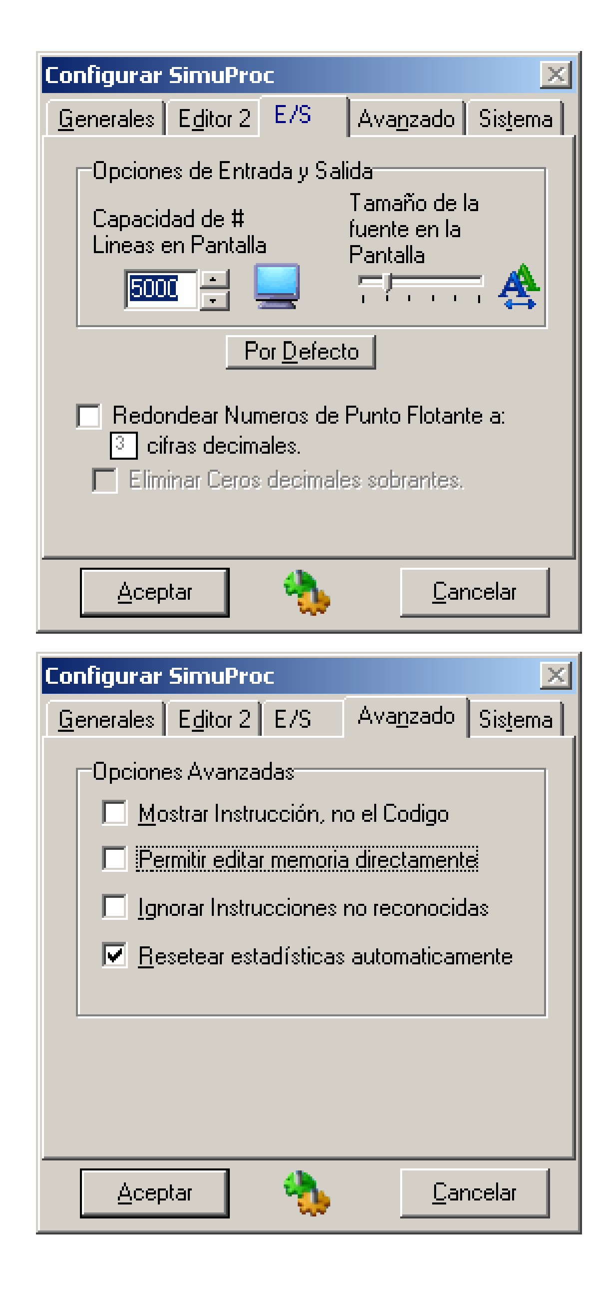


El editor permite activar o desactivar la sintaxis coloreada, facilitando la visualización de las instrucciones y comentarios del código fuente. Además de su tamaño y mostrar el numero de líneas, etc.



De la pantalla de entrada y salida también se permite cambiar los tamaños de la fuente y la cantidad de líneas que queremos recordar de las salidas de los programas.

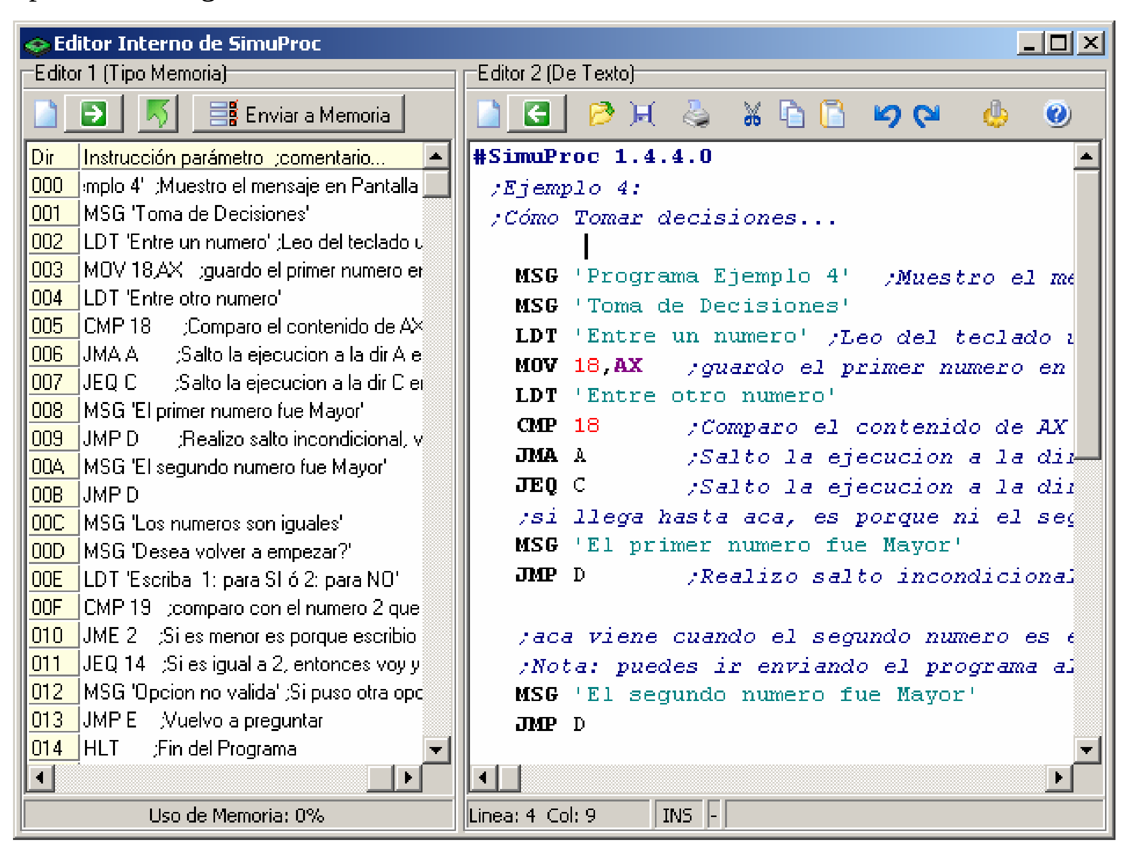
Si se requiere que los números de punto flotante hagan algún redondeo, aquí también se puede cambiar y seleccionando con cuantas cifras decimales trabajar.



Editor Interno

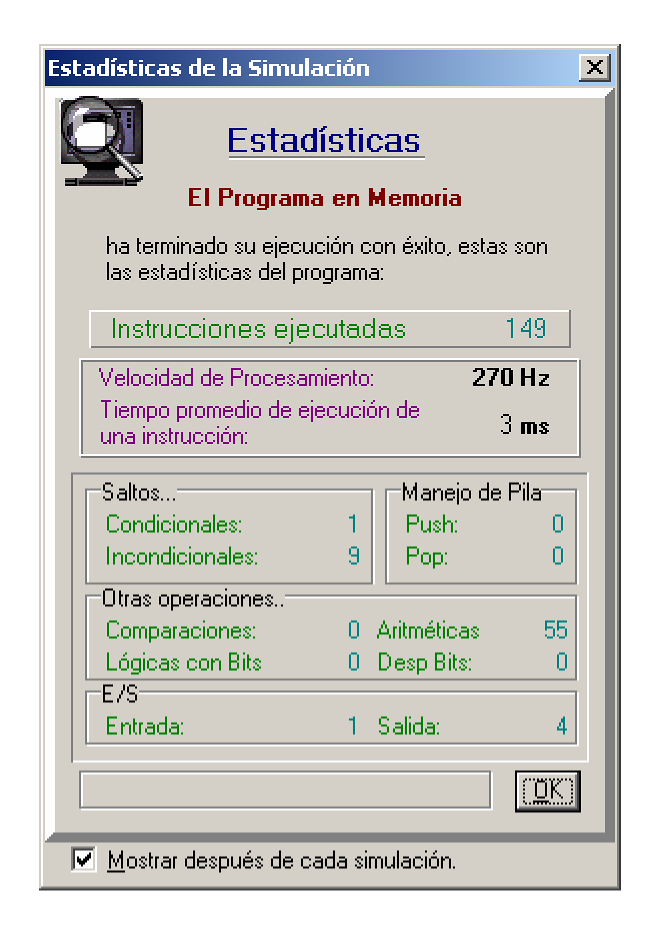
SimuProc cuenta con un Editor Interno para escribir tus programas mas fácilmente, tiene resaltado de sintaxis, soporte de Deshacer, Rehacer, Desplazamiento o tabulación de código seleccionado (presionando ctrl. Shift I ó U), marcadores

(presionando Ctrl Shift. 1-9 para marcar y ctrl. 1-9 para ir al marcador), y otras opciones configurables…



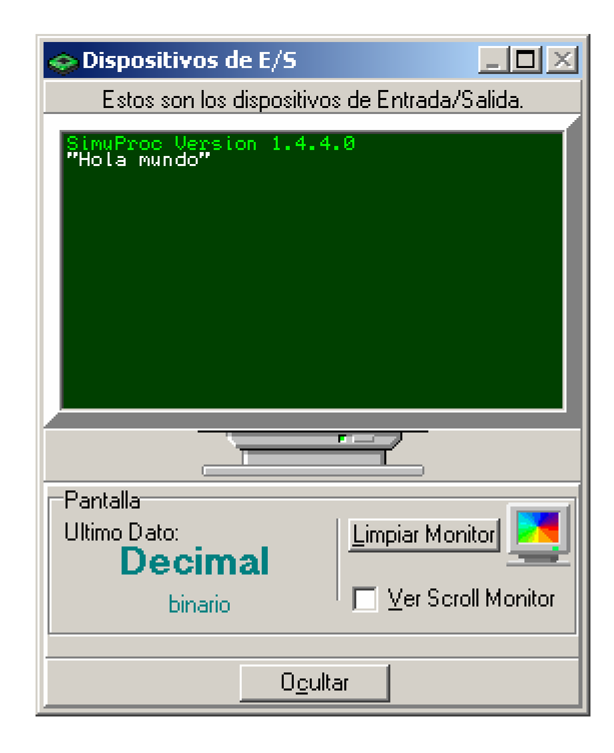
Estadísticas de la Simulación

Después o durante cada simulación se pueden mostrar estadísticas de esta, como los tipos de instrucciones que se están ejecutando, la cantidad total y una estimación de la velocidad a la cual se esta simulando.



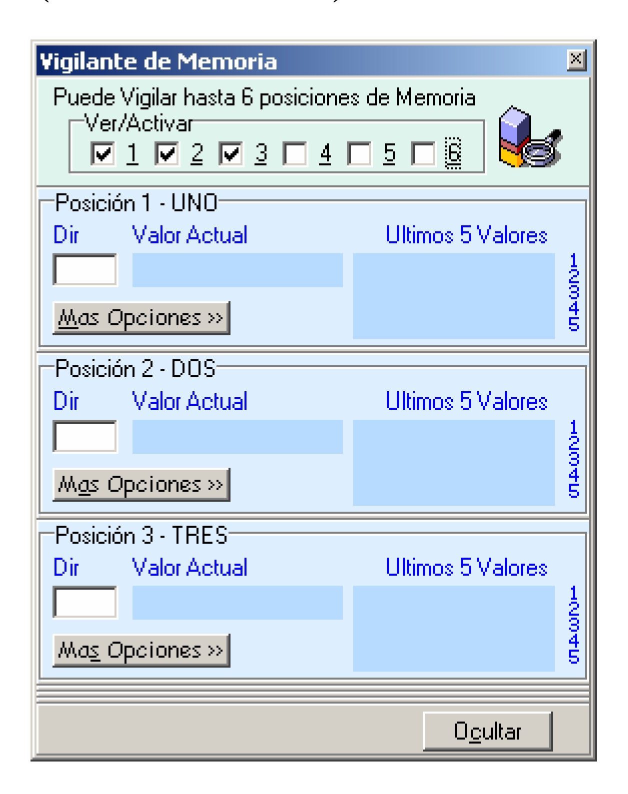
Dispositivos de Entrada/Salida

Como se mencionó anteriormente en esta pantalla se interactúa con los programas que ejecutados en el simulador, ya sea para ver la salida de estos, o para ingresar datos cuando este nos los pida.

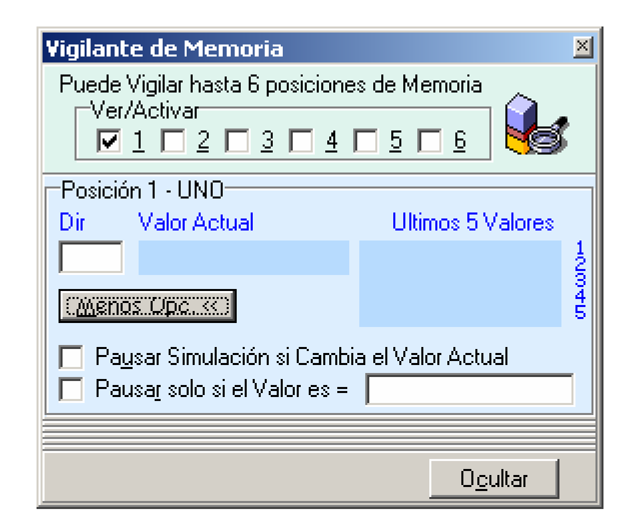


Vigilante de Memoria

El vigilante de memoria se usa cuando quiere saber los valores que ha tenido una posición de memoria (recuerda los últimos 5)

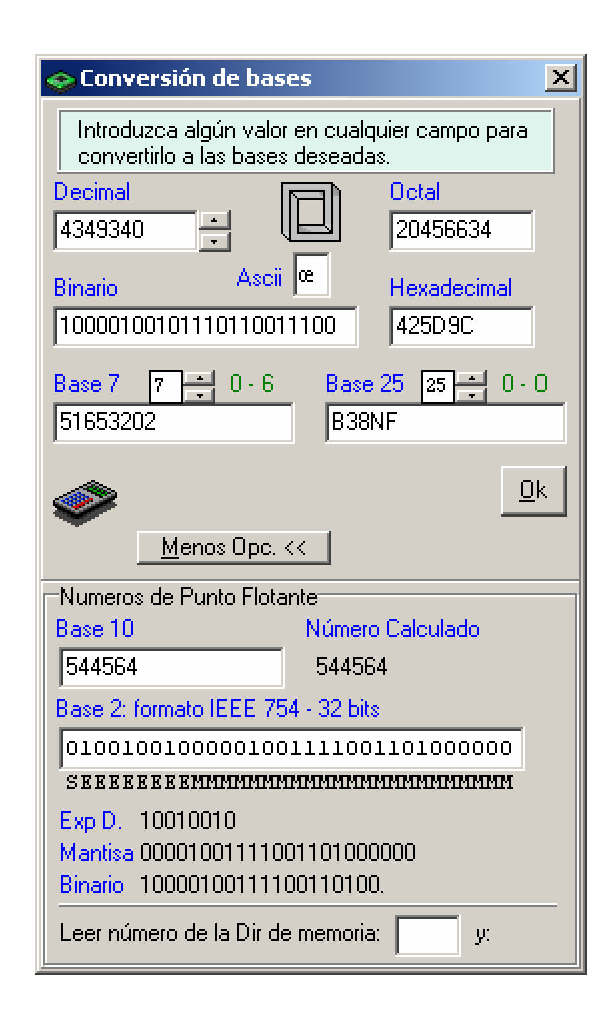


y se puede establecer alguna condición con la ejecución del programa como por ejemplo cuando un valor cambie, o sea igual a un valor predeterminado.



Conversor de Bases

Para ayudar en las conversiones rápidamente, SimuProc cuenta con un conversor de bases que soporta las bases 2 a 36 y realiza conversiones entre cualquier base, incluso conversión para números de punto flotante a IEEE 754 y viceversa


Conversión de bases
Introduzca algún valor en cualquier campo para convertirlo a las bases deseadas.
Decimal
4349340
Octal
20456634
Ascii
œ
Binario
10000100101110110011100
Hexadecimal
425D9C
Base 7
7
0 - 6
51653202
Base 25
25
0 - O
B38NF
Ok
Menos Opc. <<
Numeros de Punto Flotante
Base 10
544564
Número Calculado
544564
Base 2: formato IEEE 754 - 32 bits
01001001000001001111001101000000
SEEEEEEEEMMMMMMMMMMMMMMMMMMMMMMM
Exp D. 10010010
Mantisa 00001001111001101000000
Binario 10000100111100110100.
Leer número de la Dir de memoria:
y: